CARPATHIAN J. MATH. Online version at http://carpathian.ubm.ro
35 (2019), No. 2, 233 - 252 Print Edition: ISSN 1584 - 2851 Online Edition: ISSN 1843 - 4401

Dedicated to Prof. Juan Nieto on the occasion of his 60^{th} anniversary

Behaviour of advection-diffusion-reaction processes with forcing terms

MURAT SARI[1], SHKO ALI TAHIR[1] and ABDERRAHMAN BOUHAMIDI[2]

ABSTRACT. Without doing any linearization, this paper mainly focuses on capturing numerical behavior of the advection-diffusion-reaction (ADR) processes with forcing terms. Since the linearization of nonlinear systems loses real features, the physical systems are important to understand their natural properties. Therefore we concentrate on investigation of the real-world processes without losing their properties. To achieve the aforementioned aims, this article presents two newly combined methods; the backward differentiation formula-Spline (BDFS) and the optimal five stage and fourth-order strong stability preserving Runge-Kutta-Spline (SSPRK54S) methods. In the current methods, neither linearization nor transforming the process is required. Comparison between the two methods is carried out in dealing with the ADR problems to check the efficiency and utility of the proposed schemes. Accuracy of the methods is assessed in terms of the relative and absolute errors. The computed results showed that the BDFS method is seen to be more powerful, quite accurate and more economical in comparison with the SSPRK54S method. The current method is seen to be a very reliable alternative in solving the problem by conserving the physical properties of the nature. The BDFS method is realized to be efficient for these types of physical problems and be easy to implement. The results have revealed that the BDFS scheme is relatively free of choice of the physical parameters.

1. INTRODUCTION

The structure of the ADR equation plays an important role for describing the relation among the reaction mechanisms, convection effect and diffusion transport. They arise in various fields of science such as fluid dynamics, financial mathematics, turbulence, traffic flow, shock waves, gas dynamics etc. The ADR equation arising in various fields of science is considered as

$$\frac{\partial u}{\partial t}(x,t) = \mathscr{L}(\triangle u, \bigtriangledown u, u, x, t) + \mathscr{N}(\triangle u, \bigtriangledown u, u, x, t), \quad (x,t) \in \Omega = [a,b] \times [t_0, T]. \tag{1.1}$$

Here, $\mathscr{L}(\triangle u, \bigtriangledown u, u, x, t) = a_2 \triangle u(x,t) + a_1 \bigtriangledown u(x,t) + a_0\, u(x,t)$ is a linear partial differential operator of the second order, a_i are constant coefficients, and $\mathscr{N}$ defines a nonlinear differential part. The initial and boundary conditions are given by

$$u(x,t_0) = u_0(x), \quad u(a,t) = g_1(t), \quad u(b,t) = g_2(t), \tag{1.2}$$

where both boundary functions g_1, g_2 and initial function u_0 are known. Even though some researchers assume that the boundary functions g_1 and g_2 are differentiable, it is not necessary for all the times. In the present paper, we only assume that the boundary functions g_1 and g_2 are defined on the time interval $[t_0, T]$ without requiring the differentiability of these functions.

The generalized Burgers-Fisher equation with forcing terms (GBFEF) and the generalized Burgers-Huxley equation with forcing terms (GBHEF) can be considered to be good examples

Received: 30.09.2018. In revised form: 20.03.2019. Accepted: 27.03.2019

2010 *Mathematics Subject Classification.* 97M10, 97N50, 41A15, 35R03.

Key words and phrases. *advection-diffusion-reaction equations, spline, backward differentiation formula, optimal five stage and fourth-order strong stability preserving Runge-Kutta.*

Corresponding author: Murat Sari; sarim@yildiz.edu.tr

of the ADR equations.

The GBFEF is given by

$$\frac{\partial u}{\partial t} - \frac{1}{Re}\frac{\partial^2 u}{\partial x^2} + \beta u^{\delta}\frac{\partial u}{\partial x} - \gamma u(1-u^{\delta}) - f(x,t) = 0. \tag{1.3}$$

And similarly the GBHEF is given by

$$\frac{\partial u}{\partial t} - \frac{1}{Re}\frac{\partial^2 u}{\partial x^2} + \beta u^{\delta}\frac{\partial u}{\partial x} - \gamma u(1-u^{\delta})(u^{\delta}-C) - f(x,t) = 0, \tag{1.4}$$

where γ, β are real parameters, δ is a positive integer, $0 < Re = \dfrac{1}{\lambda} \leq 1$ and $0 < C \leq 1$.

Recently, these model equations have been considered by many researchers for both conceptual understanding of physical flows and testing various numerical methods with challenging of small and large values of the viscosity and independent parameters.

It is still crucial to do more research on finding the solution of the GBFEF with the aim of improving accuracy. Equation (1.3) was first studied by Fisher, with free of forcing term, to describe the propagation of gene in a habitat [5, 13]. The GBFEF was presented as the dynamic spread of a combustion front by Kolmogorov et al [21]. In the process of historical development, various numerical techniques to investigate the GBFE models were developed [1, 10, 12, 14, 16, 29, 33, 36, 39, 40].

The GBHEF being a nonlinear partial differential equation is of high importance for describing the interaction between diffusion and transports, convection and reaction mechanisms. There has been vast variety of numerical techniques to obtain solution of the Burgers-Huxley equation [2, 3, 4, 11, 17, 18, 19, 28, 33].

In this study, we provide two new numerical schemes, one of the two is the BDFS and the second one is the SSPRK54S. The BDFS method attempts to combine a cubic spline defined as a combination of the cubic B-splines scheme in space with the BDF scheme in time for analysing the ADR equations. The proposed methods are obtained directly from the natural spline conditions in space. The characteristics of the spline methods are continuity, smoothness and local supports [6, 8, 9, 15, 22, 23, 26, 27, 30, 31, 34]. The two schemes use collocation B-spline functions with the conditions of natural spline for space variable. The combined approaches are directly applicable to solve the ADR problems without either any linearisation or transformation process. Also these methods have additional advantages over some rival techniques such as they are relatively easy in use and are of computational cost efficiency. In this study we focus on the BDF and SSPRK54 methods for solving the resulting ordinary differential equations (ODEs) in time. The BDF method is one of the most important tool to solve differential equations. For comparison purposes, we also provide the SSPRK54 method for solving ODEs in time. Note that in this method, the SSP property guarantees the stability properties which are necessary in the numerical solutions of ODEs.

It is noticeable that, the ADR equations are highly nonlinear equations because they present the interaction between reaction, convection and diffusion mechanisms [38]. The ADR equations contain free parameters. Thus, the examination of the physical and numerical properties of the ADR equation becomes quite complex. Difficulties were experienced in the past for the small values of viscocity and the large values of δ. Since a special technique is still required to handle such problems, we propose the BDFS scheme which is a very important tool for studying the ADR problems. The proposed algorithms replace the ADR equation by an ODE system, which does not require linearization. The BDFS scheme is unconditionally stable and, the BDFS produces solutions with high-order accuracy in space and time. This paper presents a numerical comparison between the two proposed methods for solving the ADR problems. The BDFS method provides remarkable accuracy in comparison with the SSPRK54S method. Then, the results revealed that the BDFS method is more powerful than the SSPRK54S method at any value of parameters in the solution domain [7, 37]. The remainder of this paper is organized as follows. In Section 2, we give a brief introduction of the BDFS and SSPRK54S techniques using cubic natural spline interpolation to analyse the ADR equations. In Section 3, we implement the BDFS and SSPRK54S methods to handle the GBFEF and GBHEF by using the proposed methods. Then, we present numerical comparison between the two methods. In Section 4, we have presented some final remarks and future recommendations.

2. Implementation of the methods in space

For the approximate solution of the initial-boundary value problems (1.1) - (1.2), we discretize the space interval $[a,b]$ into m equal subintervals with the spatial step $h = \dfrac{b-a}{m}$. An approximation $s_h(x,t)$ to the exact solution $u(x,t)$ of (1.1) can be expressed in terms of the cubic interpolating spline in the following form

$$s_h(x,t) = \sum_{i=-1}^{m+1} \alpha_i(t) B_i(x), \tag{2.5}$$

where $\alpha_i(t)$ are unknown time dependent quantities. The cubic spline s_h interpolating the function u at the knots $x_0,\dots,x_m$ is the unique function in $\mathscr{C}^2([a,b])$ satisfying the following conditions

$$\begin{cases} s_h(x_i,t) = u(x_i,t) \quad \text{for} \quad i = 0,\dots,m, \\ \\ s_h''(a,t) = s_h''(b,t). \end{cases} \tag{2.6}$$

The interpolating cubic spline s_h is satisfying the conditions (2.6). We then have

$$s_h(x_k,t) = \sum_{i=-1}^{m+1} \alpha_i(t) B_i(x_k) = u(x_k,t),\ 0 \le k \le m, \tag{2.7}$$

with

$$s_h''(a,t) = \frac{1}{h^2}\alpha_{-1} - \frac{2}{h^2}\alpha_0 + \frac{1}{h^2}\alpha_1, \quad \text{and} \quad s_h''(b,t) = \frac{1}{h^2}\alpha_{m-1} - \frac{2}{h^2}\alpha_m + \frac{1}{h^2}\alpha_{m+1}.$$

Now, we consider the natural cubic splines which require that the second derivatives vanish at the boundaries of the interval $[a,b]$. So, the boundary conditions $s_h''(a,t) = s_h''(b,t) = 0$ lead to

$$\begin{cases} \alpha_{-1}(t) &= 2\alpha_0(t) - \alpha_1(t), \\ \alpha_{m+1}(t) &= 2\alpha_m(t) - \alpha_{m-1}(t). \end{cases} \tag{2.8}$$

By taking into account the interpolating conditions at the boundary points $x_0 = a$ and $x_m = b$, we obtain

$$s_h(x_0,t) = \frac{1}{6}\Big(\alpha_{-1}(t) + 4\alpha_0(t) + \alpha_1(t)\Big) = u(x_0,t),$$

and

$$s_h(x_m,t) = \frac{1}{6}\Big(\alpha_{m-1}(t) + 4\alpha_m(t) + \alpha_{m+1}(t)\Big) = u(x_m,t).$$

Together with the relations of (2.8) we reach

$$\alpha_0(t) = u(x_0,t) \quad \text{and} \quad \alpha_m(t) = u(x_m,t). \tag{2.9}$$

Thus, we give a description of the collocation method for the computation of the numerical solutions of a general time dependent ADR equations (1.1) with boundary and initial conditions (1.2). One can consider the following vector valued functions,

$$\mathbb{B}(x) = \begin{bmatrix} B_1(x) \\ \vdots \\ B_{m-1}(x) \end{bmatrix} \quad \text{and} \quad y(t) = \begin{bmatrix} y_1(t) \\ \vdots \\ y_{m-1}(t) \end{bmatrix} = \begin{bmatrix} \alpha_1(t) \\ \vdots \\ \alpha_{m-1}(t) \end{bmatrix}, \tag{2.10}$$

of size $(m-1)\times 1$. The function $s_h(x,t)$ and their derivatives have the following form

$$\left\{\begin{array}{rcl} s_h(x,t) &=& \alpha_{-1}(t)B_{-1}(x)+\alpha_0(t)B_0(x)+\mathbb{B}(x)^T y(t)+\alpha_m(t)B_m(x)+\alpha_{m+1}(t)B_{m+1}(x),\\ \dfrac{\partial s_h}{\partial t}(x,t) &=& \alpha'_{-1}(t)B_{-1}(x)+\alpha'_0(t)B_0(x)+\mathbb{B}(x)^T y'(t)+\alpha'_m(t)B_m(x)+\alpha'_{m+1}(t)B_{m+1}(x),\\ \dfrac{\partial s_h}{\partial x}(x,t) &=& \alpha_{-1}(t)B'_{-1}(x)+\alpha_0(t)B'_0(x)+\mathbb{B}'(x)^T y(t)+\alpha_m(t)B'_m(x)+\alpha_{m+1}(t)B'_{m+1}(x),\\ \dfrac{\partial^2 s_h}{\partial x^2}(x,t) &=& \alpha_{-1}(t)B''_{-1}(x)+\alpha_0(t)B''_0(x)+\mathbb{B}''(x)^T y(t)+\alpha_m(t)B''_m(x)+\alpha_{m+1}(t)B''_{m+1}(x). \end{array}\right. \tag{2.11}$$

The collocation method consists of substituting u and its derivatives in (1.1) by the expression s_h

TABLE 1. B-spline values and its derivatives at points x_i.

x	x_{i-2}	x_{i-1}	x_i	x_{i+1}	x_{i+2}
$B_i(x)$	0	$1/6$	$4/6$	$1/6$	0
$B'_i(x)$	0	$-1/2h$	0	$1/2h$	0
$B''_i(x)$	0	$1/h^2$	$-2/h^2$	$1/h^2$	0

and its derivatives given by (2.11) and the values given in Table 1. So, by evaluating these equations at points x_i for $i=0,\dots,m$, we get the following relations. For the end points x_0 and x_m, we have

$$\frac{\partial s_h}{\partial t}(x_0,t)=a_2\frac{\partial^2 s_h}{\partial x^2}(x_0,t)+a_1\frac{\partial s_h}{\partial x}(x_0,t)+a_0 s_h(x_0,t)+F(y(t),x_0,t), \tag{2.12}$$

and

$$\frac{\partial s_h}{\partial t}(x_m,t)=a_2\frac{\partial^2 s_h}{\partial x^2}(x_m,t)+a_1\frac{\partial s_h}{\partial x}(x_m,t)+a_0 s_h(x_m,t)+F(y(t),x_m,t), \tag{2.13}$$

where F is the function representing the nonlinear part. Taking into account of the relations (2.8), (2.9), (2.12) and (2.13), we obtain

$$\begin{array}{rcl} \alpha'_0(t) &=& \big(a_0+\dfrac{a_1}{h}\big)g_1(t)-\dfrac{a_1}{h}\alpha_1(t)+F(y(t),x_0,t),\\ \alpha'_m(t) &=& \big(a_0-\dfrac{a_1}{h}\big)g_2(t)+\dfrac{a_1}{h}\alpha_{m-1}(t)+F(y(t),x_m,t). \end{array} \tag{2.14}$$

Now, from (2.11) and (2.14), by evaluating the equation at points x_1 and x_{m-1}, we get

$$\begin{array}{l} \mathbb{B}^{\mathrm{T}}(x_1)y'(t)=\big(\dfrac{2a_0}{3}+\dfrac{a_1}{6h}-\dfrac{2a_2}{h^2}\big)y_1(t)+\big(\dfrac{a_0}{6}-\dfrac{a_1}{2h}+\dfrac{a_2}{h^2}\big)y_2(t)\\ \qquad +\big(\dfrac{a_2}{h^2}+\dfrac{a_1}{3h}\big)g_1(t)+F(y(t),x_1,t)-\dfrac{1}{6}F(y(t),x_0,t), \end{array} \tag{2.15}$$

and

$$\begin{array}{l} \mathbb{B}^{\mathrm{T}}(x_{m-1})y'(t)=\big(\dfrac{a_0}{6}+\dfrac{a_1}{2h}+\dfrac{a_2}{h^2}\big)y_{m-2}(t)+\big(\dfrac{2a_0}{3}-\dfrac{a_1}{6h}-\dfrac{2a_2}{h^2}\big)y_{m-1}(t)\\ \qquad +\big(\dfrac{a_2}{h^2}-\dfrac{a_1}{3h}\big)g_2(t)+F(y(t),x_{m-1},t)-\dfrac{1}{6}F(y(t),x_m,t). \end{array} \tag{2.16}$$

At points x_i, $i=2,\dots,m-2$, we obtain

$$\mathbb{B}^{\mathrm{T}}(x_i)y'(t)=\Big(a_2\mathbb{B}''(x_i)+a_1\mathbb{B}'(x_i)+a_0\mathbb{B}(x_i)\Big)^{\mathrm{T}}y(t)+F(y(t),x_i,t). \tag{2.17}$$

For $i = 2, \ldots, m-2$, we then have

$$a_2\mathbb{B}''(x_i) + a_1\mathbb{B}'(x_i) + a_0\mathbb{B}(x_i) = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \dfrac{a_0}{6} + \dfrac{a_1}{2h} + \dfrac{a_2}{h^2} \\ \dfrac{2a_0}{3} - \dfrac{2a_2}{h^2} \\ \dfrac{a_0}{6} - \dfrac{a_1}{2h} + \dfrac{a_2}{h^2} \\ 0 \\ \vdots \\ 0 \end{pmatrix} \begin{array}{l} \\ \\ \\ \longrightarrow i-1 \\ \longrightarrow i \\ \longrightarrow i+1. \\ \\ \\ \\ \end{array} \tag{2.18}$$

The approximating cubic spline s_h must also satisfy the initial conditions (1.2) at points $x_0, \ldots, x_m$ and at initial time t_0 :

$$\begin{cases} s_h(x_0, t_0) &= u_0(x_0), \quad \text{for} \quad i = 0, \\ s_h(x_i, t_0) &= u_0(x_i), \quad \text{for} \quad i = 1, ..., m-1, \\ s_h(x_m, t_0) &= u_0(x_m), \quad \text{for} \quad i = m. \end{cases} \tag{2.19}$$

By using the expressions (2.8), (2.9) and (2.11), we find out the condition

$$Ay(t_0) = y_0, \tag{2.20}$$

where y_0 is the vector given by $y_0 = [u_0(x_1) - \frac{1}{6}u_0(x_0), u_0(x_2), \ldots, u_0(x_{m-2}), u_0(x_{m-1}) - \frac{1}{6}u_0(x_m)]^T$. The matrix A of size $(m-1) \times (m-1)$ is given by

$$A = \frac{1}{6}\begin{pmatrix} 4 & 1 & 0 & \cdots & 0 \\ 1 & 4 & 1 & & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & 1 & 4 & 1 \\ 0 & \cdots & 0 & 1 & 4 \end{pmatrix}. \tag{2.21}$$

Now, equations (2.15) - (2.20) are expressed as the following system of ordinary differential equations

$$\begin{cases} A\dfrac{dy(t)}{dt} &= Dy(t) + \Phi(y(t)), \\ Ay(t_0) &= y_0, \end{cases} \tag{2.22}$$

where the matrix D of size $(m-1) \times (m-1)$ is

$$D = \begin{pmatrix} d_0 + \dfrac{a_1}{6h} & d_1 & 0 & \cdots & 0 \\ d_1' & d_0 & d_1 & & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & d_1' & d_0 & d_1 \\ 0 & \cdots & 0 & d_1' & d_0 - \dfrac{a_1}{6h} \end{pmatrix}, \tag{2.23}$$

with $d_0 = \frac{2a_0}{3} - \frac{2a_2}{h^2}$, $d_1 = \frac{a_0}{6} - \frac{a_1}{2h} + \frac{a_2}{h^2}$, and $d_1' = \frac{a_0}{6} + \frac{a_1}{2h} + \frac{a_2}{h^2}$. The vector valued function Φ is given by $\Phi(y(t)) = [\Phi_1(y(t)), \Phi_2(y(t)), \ldots, \Phi_{m-2}(y(t)), \Phi_{m-1}(y(t))]^T$ with

$$\Phi_1(y(t)) = \left(\frac{a_2}{h^2} + \frac{a_1}{3h}\right) g_1(t) + F(y(t), x_1, t) - \frac{1}{6} F(y(t), x_0, t),$$

$$\Phi_{m-1}(y(t)) = \left(\frac{a_2}{h^2} - \frac{a_1}{3h}\right) g_2(t) + F(y(t), x_{m-1}, t) - \frac{1}{6} F(y(t), x_m, t),$$

and

$$\Phi_i(y(t)) = F(y(t), x_i, t) \text{ for } i = 2, \ldots, m-2.$$

This section begins with a brief study of the spline methods for the numerical solution of the ADR equations in space. Then, we will present the BDF and SSPRK54 methods and will apply to solve the ADR problems in time.

3. Implementation of the methods in time

3.1. **The BDF method.** Backward differentiation formulae (BDF) are implicit multi-step methods for numerically solving the initial-value problems of the form (2.22). They are the most widely used methods for solving ODEs due to their stability properties. In addition, the BDF formulae are based on numerical differentiation. The time interval $[t_0, T]$ is divided into N subintervals with the time step $\Delta t = \frac{T - t_0}{N}$ with the knots $t_n = t_0 + n\Delta t$ for $n = 0, \ldots, N$. The BDF method applied to (2.22) gives rise to the following approximations

$$Ay_n - \beta h\big[Dy_n + \Phi(y_n)\big] - \sum_{j=0}^{p} \eta_j A y_{n-j} = 0, \tag{3.24}$$

where $y_n = [y_{1,n}, \ldots, y_{m-1,n}]^T$ is an approximation obtained by the BDF method. Here vector $y(t)$ is given by (2.10) at $t = t_n$. The coefficients η_j, β are given in Table 2 for the p-step BDF formula. At each time step n, we have to solve for y_n, equation (3.24) by rearranging it in the following form

TABLE 2. Coefficients of the BDF p-step method for $p = 6$.

p	β	η_0	η_1	η_2	η_3	η_4	η_5	η_6
1	1	-1						
2	$\frac{2}{3}$	$\frac{4}{3}$	$\frac{-1}{3}$					
3	$\frac{6}{11}$	$\frac{18}{11}$	$\frac{-9}{11}$	$\frac{2}{11}$				
4	$\frac{22}{25}$	$\frac{48}{25}$	$\frac{-36}{25}$	$\frac{16}{25}$	$\frac{-3}{25}$	$\frac{3}{25}$		
5	$\frac{60}{137}$	$\frac{300}{137}$	$\frac{-300}{137}$	$\frac{200}{137}$	$\frac{-75}{137}$	$\frac{-12}{137}$	$\frac{-12}{137}$	
6	$\frac{60}{137}$	$\frac{300}{137}$	$\frac{-300}{137}$	$\frac{200}{137}$	$\frac{-75}{137}$	$\frac{-72}{147}$	$\frac{-75}{147}$	$\frac{10}{147}$

$$\mathscr{G}(y_n) = (A - \eta_0 I)y_n - \beta h\big[Dy_n + \Phi(y_n)\big] - \sum_{j=1}^{p} \eta_j A y_{n-j} = 0, \tag{3.25}$$

where I is an $(m-1)\times(m-1)$ identity matrix. Equation (3.25) can efficiently be solved by using the Newton method with the initial value taken from the last time step. Here, the Newton method for the approximation of y_n generates iterations (ξ_k) given by

$$\begin{cases} \xi_0 & \\ \xi_{k+1} = \xi_k - [J_{\mathscr{G}}(\xi_k)]^{-1}\mathscr{G}(\xi_k), & k \geqslant 0 \end{cases} \tag{3.26}$$

where $J_{\mathscr{G}}(\xi_k)$ is the Jacobian matrix of $\mathscr{G}$ at point ξ_k. We have

$$J_{\mathscr{G}}(\xi_k) = (A-\eta_0 I) - \beta h(D + J_\Phi(\xi_k)), \tag{3.27}$$

with J_Φ being the Jacobian matrix of Φ. The value of the interpolating spline s_h given by (2.5) at time t_n is

$$s_h(x,t_n) = \alpha_{-1}(t_n)B_{-1}(x) + \alpha_0(t_n)B_0(x) + \mathbb{B}(x)^T y(t) + \alpha_m(t_n)B_m(x) + \alpha_{m+1}(t_n)B_{m+1}(x).$$

To simplify the proposed method, we ignore the error of the Newton method. Then we approximate $\alpha_i(t_n)$ by $\widehat{\alpha}_{i,n}$ as given by

$$\begin{array}{lclcl} \widehat{\alpha}_{i,n} & = & y_{i,n}, \quad i=1,\ldots,m-1, & & \\ \widehat{\alpha}_{0,n} & = & u(x_0,t_n) & = & g_1(t_n), \\ \widehat{\alpha}_{m,n} & = & u(x_m,t_n) & = & g_2(t_n), \\ \widehat{\alpha}_{-1,n} & = & 2\widehat{\alpha}_{0,n} - y_{1,n} & = & 2g_1(t_n) - y_{1,n}, \\ \widehat{\alpha}_{m+1,n} & = & 2\widehat{\alpha}_{m,n} - y_{m-1,n} & = & 2g_2(t_n) - y_{m-1,n}. \end{array} \tag{3.28}$$

Here, the value $s_h(x,t_n)$ of the approximation spline s_h given by (2.5) at time t_n for $n=0,\ldots,N$ are approximated by the values $\widehat{s}_{n,h}(x)$ where $\widehat{s}_{n,h}$ be the cubic spline given in the form

$$\widehat{s}_{n,h}(x) = \sum_{i=-1}^{m+1} \widehat{\alpha}_{i,n} B_i(x).$$

We have then $s_h(x,t_n) \backsimeq \widehat{s}_{n,h}(x)$ for all $x \in [a,b]$.

3.2. **The SSPRK54 method.** Now, we present the SSPRK54S methods to numerically approximate the solution of the ODE (2.22). The SSPRK54 method has order at most four. However, we pay attention to the optimal five-stage, fourth order method [35]. The time interval $[t_0,T]$ is divided into N subintervals as previously mentioned. At each time step n, we have to solve y_n of equation (2.22) and rearrange it in the following form

$$\mathscr{F}(y_n) = (A-I)y_n - \big[Dy_n + \Phi(y_n)\big] = 0, \tag{3.29}$$

where I is an $(m-1)\times(m-1)$ identity matrix thus,

$$\begin{aligned} y_1 &= y_n + 0.391752226571890\Delta(t)\mathscr{F}(y_n) \\ y_2 &= 0.444370493651235y_n + 0.555629506348765y_1 + 0.368410593050371\Delta(t)\mathscr{F}(y_1) \\ y_3 &= 0.620101851488403y_n + 0.379898148511597y_2 + 0.251891774271694\Delta(t)\mathscr{F}(y_2) \\ y_4 &= 0.178079954393132y_n + 0.821920045606868y_3 + 0.544974750228521\Delta(t)\mathscr{F}(y_3) \\ y_{n+1} &= 0.517231671970585y_2 + 0.096059710526147y_3 + 0.063692468666290\Delta(t)\mathscr{F}(y_3) \\ &+ 0.386708617503269\Delta(t)\mathscr{F}(y_4) + 0.226007483236906\Delta(t)\mathscr{F}(y_4). \end{aligned}$$

4. Illustrative Examples

In this section, we present some numerical results computed by the BDFS and SSPRK54S methods. The efficiency and accuracy of the proposed methods have been tested for different cases. We have discretized the solution domain $[a,b]$ using new equally spaced points $x'_i = a + i\frac{b-a}{k}$, for $i = 0,\dots,k$. It is important to note that the produced solutions are not presented only at the grid points but also at optional points in the solution domain. In order to measure the accuracy of the proposed schemes, the relative errors e_1, e_2 and e_∞ are defined by

$$k \longrightarrow e_1(k) = \frac{\|U-S_k\|_1}{\|U\|_1} = \frac{\sum_{n=0}^{N}\left(\sum_{i=0}^{k} |u(x'_i,t_n) - s_{h,n}(x'_i)|\right)}{\sum_{n=0}^{N}\left(\sum_{n=0}^{N} |u(x'_i,t_n)|\right)}, \tag{4.30}$$

$$k \longrightarrow e_2(k) = \frac{\|U-S_k\|_2}{\|U\|_2} = \frac{\sqrt{\sum_{n=0}^{N}\left(\sum_{i=0}^{k} |u(x'_i,t_n) - s_{h,n}(x'_i)|^2\right)}}{\sqrt{\sum_{n=0}^{N}\left(\sum_{n=0}^{N} |u(x'_i,t_n)|^2\right)}}, \tag{4.31}$$

$$k \longrightarrow e_\infty(k) = \frac{\|U-S_k\|_\infty}{\|U\|_\infty} = \frac{\max_{0\le n\le N}\left(\max_{0\le i\le k} |u(x'_i,t_n) - s_{h,n}(x'_i)|\right)}{\max_{0\le n\le N}\left(\max_{0\le i\le k} |u(x'_i,t_n)|\right)}, \tag{4.32}$$

where $U = (u(x'_i,t_n))$ and $S_k = (s_{h,n}(x'_i))$ are the matrices of size $(N+1)\times(N+1)$ whose entries are the values of the exact and the numerical solutions, respectively, at the points (x'_i,t_n) with step size $h = \frac{b-a}{k}$ and time step size $\Delta t = \frac{T-t_0}{N}$. Here, the produced numerical solutions are a set of points x'_i which are different from the set of knots on the B-spline discretizations. All computations have been carried out by using MATLAB 2018 on a PC with 16 significant decimal digits.

4.1. **Numerical Solutions of the GBFEF.** Consider the GBFEF of the form

$$\frac{\partial u}{\partial t} - \frac{1}{Re}\frac{\partial^2 u}{\partial x^2} + \beta u^\delta \frac{\partial u}{\partial x} - \gamma u(1-u^\delta) - f(x,t) = 0, \quad (x,t)\in\Omega = [a,b]\times[t_0,T], \tag{4.33}$$

with the initial and boundary conditions given by

$$u(x,t_0) = u_0(x), \tag{4.34}$$

$$u(a,t) = g_1(t), \quad u(b,t) = g_2(t), \tag{4.35}$$

where $Re = \frac{1}{\lambda}$ is the Reynolds number of the viscous fluid flow problem, the functions g_1, g_2 and the initial function u_0 are known. The structure of the GBFE can be seen as a useful model for describing the relation between the reaction mechanisms, convection effect and diffusion transport. It also arises in various fields such as financial mathematics, turbulence, fluid mechanics, traffic flow, shock waves and gas dynamics.

In this example, the presented numerical schemes in solving (4.33) are to find an approximation $s_h(x,t)$ to the exact solution $u(x,t)$ given in equation (2.5). By rearranging equation (4.33) as the form of (1.1), we obtain linear part and nonlinear part (involving the forcing term), respectively as

$$\mathscr{L}(\frac{\partial^2 u}{\partial x^2}, \frac{\partial u}{\partial x}, u, x, t) = \lambda u_{xx} + \gamma u,$$

and

$$\mathscr{N}(\frac{\partial^2 u}{\partial x^2}, \frac{\partial u}{\partial x}, u, x, t) = -\delta u^\delta u_x - \gamma u u^\delta + f(x,t).$$

Now, $a_0 = \gamma$, $a_1 = 0$ and $a_2 = \lambda$. Considering the relations (2.14), we obtain

$$\begin{aligned} y_0'(t) &= \gamma g_1(t) + F(y(t), x_0, t), \\ y_m'(t) &= \gamma g_2(t) + F(y(t), x_m, t), \end{aligned} \tag{4.36}$$

where

$$F(y(t),x_0,t) = \big(1-(g_1(t))^\delta\big) - \frac{\beta}{h}(g_1(t))^\delta\big(g_1(t)-y_1(t)\big) + f(x_0,t),$$

and

$$F(y(t),x_m,t) = \big(1-(g_2(t))^\delta\big) - \frac{\beta}{h}(g_2(t))^\delta\big(y_{m-1}(t)-g_2(t)\big) + f(x_m,t).$$

Thus, by evaluating equations (2.11) and (4.36) at points x_1 and x_{m-1}, we find

$$\frac{4}{6}y_1'(t) + \frac{1}{6}y_2'(t) = (\frac{2\gamma}{3} - \frac{2\lambda}{h^2})y_1(t) + (\frac{\gamma}{6} + \frac{\lambda}{h^2})y_2(t) + (\frac{\lambda}{h^2})g_1(t) + F(y(t),x_1,t) - \frac{1}{6}F(y(t),x_0,t), \tag{4.37}$$

and

$$\begin{aligned} \frac{1}{6}y_{m-2}'(t) + \frac{4}{6}y_{m-1}'(t) &= (\frac{\gamma}{6} + \frac{\lambda}{h^2})y_{m-2}(t) + (\frac{2\gamma}{3} - \frac{2\lambda}{h^2})y_{m-1}(t) + (\frac{\lambda}{h^2})g_2(t) \\ &+ F(y(t),x_{m-1},t) - \frac{1}{6}F(y(t),x_m,t). \end{aligned} \tag{4.38}$$

At points x_i, $i = 2,\dots,m-2$, we give

$$\frac{1}{6}y_{i-1}' + \frac{4}{6}y_i' + \frac{1}{6}y_{i+1}' = (\frac{\gamma}{6} + \frac{\lambda}{h^2})y_{i-1}(t) + (\frac{2\gamma}{3} - \frac{2\lambda}{h^2})y_i(t) + (\frac{\gamma}{6} - \frac{\lambda}{h^2})y_{i+1}(t) + F(y(t),x_i,t)\ , \tag{4.39}$$

where

$$\begin{aligned} F(y(t),x_1,t) &= -\frac{\beta}{2h}\big(\frac{1}{6}g_1(t) + \frac{4}{6}y_1(t) + \frac{1}{6}y_2(t)\big)^\delta\big(g_1(t)-y_2(t)\big) \\ &- \frac{\gamma}{6}\big(g_1(t)+4y_1(t)+\alpha_2(t)\big)\big(\frac{1}{6}g_1(t) + \frac{4}{6}y_1(t) + \frac{1}{6}y_2(t)\big)^\delta\big) + f(x_1,t), \\ F(y(t),x_{m-1},t) &= -\frac{\beta}{2h}\big(\frac{1}{6}y_{m-2}(t) + \frac{4}{6}y_{m-1}(t) + \frac{1}{6}g_2(t)\big)^\delta\big(y_{m-2}(t)-g_2(t)\big) \\ &- \big(\frac{1}{6}y_{m-2}(t) + \frac{4}{6}y_{m-1}(t) + \frac{1}{6}g_2(t)\big)^\delta\big) + f(x_{m-1},t). \\ F(y(t),x_i,t) &= -\frac{\beta}{2h}\big(\frac{1}{6}y_{i-1}(t) + \frac{4}{6}y_i(t) + \frac{1}{6}y_{i+1}(t)\big)^\delta\big(y_{i-1}(t)-y_{i+1}(t)\big) \\ &- \frac{\gamma}{6}\big(y_{i-1}(t)+4y_i(t)+y_{i+1}(t)\big) - \big(\frac{1}{6}y_{i-1}(t) + \frac{4}{6}y_i(t) + \frac{1}{6}y_{i+1}(t)\big)^\delta + f(x_i,t)\ . \end{aligned}$$

The approximating cubic spline s_h must also satisfy the initial condition (4.34) at points $x_0,\dots,x_m$ and at initial time t_0 :

$$\begin{cases} s_h(x_0,t_0) &= u_0(x_0), \quad \text{for} \quad i=0,\\ s_h(x_i,t_0) &= u_0(x_i), \quad \text{for} \quad i=1,...,m-1,\\ s_h(x_m,t_0) &= u_0(x_m), \quad \text{for} \quad i=m. \end{cases} \tag{4.40}$$

By virtue of (2.8), (2.9) and (2.11), we obtain

$$A\alpha(t_0)=\alpha_0, \tag{4.41}$$

where $y_0=[6u_0(x_1)-g_1(t_0),6u_0(x_2),\dots,6u_0(x_{m-2}),6u_0(x_{m-1})-g_2(t_0)]^T$ and the matrix A of size $(m-1)\times(m-1)$ is given by (2.21). Now, equations (4.37), (4.38), (4.39) and (4.41) are expressed as the following system of ordinary differential equations

$$\begin{cases} A\dfrac{dy(t)}{dt} &= Dy(t)+\Phi(y(t)),\\ \\ Ay(t_0) &= y_0. \end{cases} \tag{4.42}$$

The matrix D of size $(m-1)\times(m-1)$ is

$$D=\begin{pmatrix} d_0 & d_1 & 0 & \cdots & 0\\ d_1' & d_0 & d_1 & & \vdots\\ 0 & \ddots & \ddots & \ddots & 0\\ \vdots & & d_1' & d_0 & d_1\\ 0 & \cdots & 0 & d_1' & d_0 \end{pmatrix}, \tag{4.43}$$

where $d_0=\dfrac{2\gamma}{3}-\dfrac{2\lambda}{h^2}$, $d_1=\dfrac{\gamma}{6}+\dfrac{\lambda}{h^2}$ and $d_1'=\dfrac{\gamma}{6}+\dfrac{\lambda}{h^2}$. The vector valued function Φ is given by

$$\Phi(y(t))=[\Phi_1(y(t)),\Phi_2(y(t)),\dots,\Phi_{m-2}(y(t)),\Phi_{m-1}(y(t))]^T,$$

where

$$\Phi_1(y(t))=\big(\frac{\lambda}{h^2}\big)g_1(t)+F(y(t),x_1,t)-\frac{1}{6}F(y(t),x_0,t),$$
$$\Phi_{m-1}(y(t))=\big(\frac{\lambda}{h^2}\big)g_2(t)+F(y(t),x_{m-1},t)-\frac{1}{6}F(y(t),x_m,t),$$

and

$$\Phi_i(y(t))=F(y(t),x_i,t) \ \text{ for } \ i=2,\dots,m-2.$$

Now, we solve the system (4.42) by using the BDFS and SSPRK54S methods.

4.1.1. **Example 1.** We consider, equation (4.33) in the following form

$$u_t-\lambda u_{xx}+\beta u^{\delta}u_x-\gamma u(1-u^{\delta})=0, \tag{4.44}$$

with the intial condition

$$u(x,0)=\big(\frac{1}{2}+\frac{1}{2}tanh\big(\big(\frac{-\beta\delta}{2(\delta+1)}\big)x\big)\big)^{1/\delta}=u_0(x), \tag{4.45}$$

and the boundary conditions

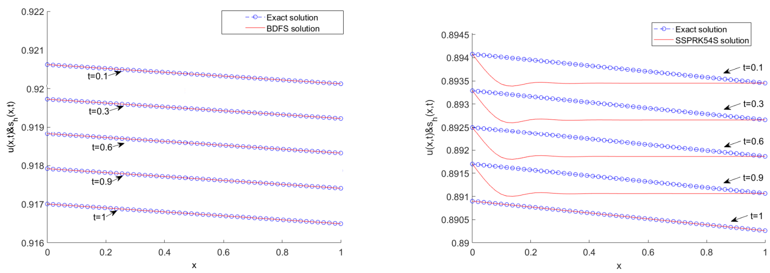
$$u(0,t)=\big(\frac{1}{2}+\frac{1}{2}tanh\big[\frac{-\beta\delta}{2(\delta+1)}\big(-\big(\frac{\beta}{\delta+1}+\frac{\gamma(\delta+1)}{\beta}\big)\big)t\big]\big)^{1/\delta}=g_1(t), \tag{4.46}$$

$$u(1,t)=\big(\frac{1}{2}+\frac{1}{2}tanh\big[\frac{-\beta\delta}{2(\delta+1)}\big(1-\big(\frac{\beta}{\delta+1}+\frac{\gamma(\delta+1)}{\beta}\big)\big)t\big]\big)^{1/\delta}=g_2(t). \tag{4.47}$$

Exact solution of equation (4.44) is given by

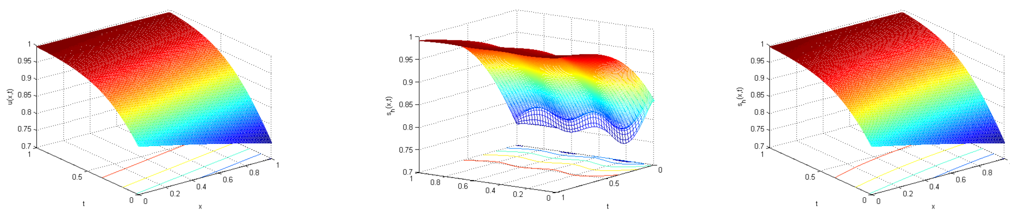
$$u(x,t)=\Big(\frac{1}{2}+\frac{1}{2}tanh\Big[\frac{-\beta\delta}{2(\delta+1)}\Big(x-\Big(\frac{\beta}{\delta+1}+\frac{\gamma(\delta+1)}{\beta}\Big)\Big)t\Big]\Big)^{1/\delta}=g_2(t). \tag{4.48}$$

This problem is solved in the domain $(x,t)\in\Omega=[-1,1]\times[0,1]$ with various values of parameters



(A) BDFS and exact solutions (B) SSPRK54S and exact solutions

FIGURE 1. Computed solutions for $\lambda=1$, $\beta=\gamma=0.01$, $\delta=8$, $h=0.002$ and $\Delta t=1e-03$



(A) Exact solution (B) SSPRK54S solution (C) BDFS solution

FIGURE 2. Computed solutions for the parameters $\lambda=0.0001$, $\gamma=1$, $\beta=0.1$, $\delta=4$ and $\Delta t=1e-03$

TABLE 3. The relative errors of the proposed methods for Example 1

(A)

$\lambda=0.01$, $\delta=500$, $\beta=\gamma=0.01$, $\Delta t=1e-4$		
Errors	SSPRK54S	BDFS
e_1	N.W.	$4.03e-7$
e_2	N.W.	$4.13e-7$
e_∞	N.W.	$3.82e-6$

(B)

$\lambda=0.0005$, $\delta=10000$, $\beta=\gamma=1$, $\Delta t=1e-4$		
Errors	SSPRK54S	BDFS
e_1	N.W.	$2.29e-8$
e_2	N.W.	$2.01e-8$
e_∞	N.W.	$6.97e-7$

λ, δ, β, γ and Δt by the proposed methods. In Table 3, the relative errors are presented for various values of parameters.The BDFS results are still very accurate while the SSPRK54S did not work (N.W.) for larger values of δ. Also, the BDFS results showed that the relative errors decrease as the parameter δ increases. The relative and absolute errors for the computation are presented for different time levels and $\delta=1$, 4, 40, 50 in Table 4 and compared with those available in the literature. It is concluded from the comparison of the results in these tables that the proposed schemes are

TABLE 4. Comparisons of the errors for Example 1

(A)

$\lambda=\delta=1,\ \beta=\gamma=0.001$					
	SSPRK54S	BDFS	[20]	[24]	[25]
Errors	$\Delta t=1e-2$		$\Delta t=1e-4$		
e_1	$3.72e-4$	$2.38e-3$	- - -	- - -	- - -
e_2	$3.78e-4$	$4.15e-4$	- - -	- - -	- - -
e_∞	$4.64e-4$	$3.41e-4$	- - -	- - -	- - -
L_∞	$5.93e-9$	$5.82e-8$	$1.93e-5$	$6.44e-7$	$1.22e-9$

(B)

$\delta=4,\ \lambda=1,\ \Delta t=1e-4$					
	SSPRK54S	BDFS	[24]	[25]	[40]
Errors	$\beta=-0.01,\gamma=1$		$\beta=1,\gamma=0.5$		
e_1	$1.07e-2$	$1.55e-4$	- - -	- - -	- - -
e_2	$1.16e-2$	$5.85e-5$	- - -	- - -	- - -
e_∞	$1.43e-2$	$5.64e-5$	- - -	- - -	- - -
L_∞	$2.03e-5$	$4.82e-8$	$1.22e-5$	$1.08e-8$	$1.44e-6$

(C)

$\lambda=1,\ \Delta t=1e-4$					
	SSPRK54S	BDFS	[20]	[24]	[25]
Errors	$\delta=50,\beta=\gamma=0.001$		$\delta=2,\beta=\gamma=1$		
e_1	$1.67e-3$	$1.02e-5$	- - -	- - -	- - -
e_2	$3.52e-4$	$4.15e-7$	- - -	- - -	- - -
e_∞	$1.33e{-}4$	$6.34e-7$	- - -	- - -	- - -
L_∞	$3.07e-7$	$3.37e-9$	$2.5e-4$	$2.1e-6$	$1.7e-7$

(D)

$\lambda=1,\ \beta=0\ \Delta t=1e^{-4}$					
	SSPRK54S	BDFS	[24]	[25]	[29]
Errors	$\delta=40,\gamma=0.001$		$\delta=8,\gamma=1$		
e_1	$2.47e-5$	$3.88e-7$	- - -	- - -	- - -
e_2	$2.82e-5$	$4.01e-7$	- - -	- - -	- - -
e_∞	$1.83e-5$	$3.94e-10$	- - -	- - -	- - -
L_∞	$4.85e-9$	$2.03e-17$	$1.19e-11$	$5.5e-16$	$3.6e-11$

very accurate for all values of $\delta>0$. We have depicted the BDFS, SSPRK54S solutions and exact solution for various time values in Figures 1(A) and 1(B). It can be seen from the results that the BDFS is more accurate than the SSPRK54S. The physical behavior of the solutions is illustrated in Figure 2. In conclusion, the computed results in the present example, show that, the BDFS method has no restriction on the choice of parameters.

4.1.2. **Example 2.** Consider the GBFE with an external force $f(x,t)$

$$u_t-\lambda u_{xx}+\beta u^\delta u_x-\gamma u(1-u^\delta)=f(x,t), \tag{4.49}$$

in the domain $[-1,1]\times[-1,1]$ with the Dirichlet boundary conditions and the initial condition, given by

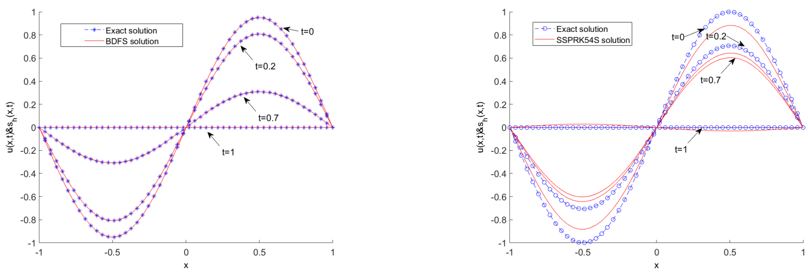
$$\left\{\begin{array}{lcl} u(0,t) & = & 0 \\ u(1,t) & = & 0 \\ u(x,0) & = & 0. \end{array}\right. \tag{4.50}$$

We choose the external force as

$$\begin{aligned} f(x,t) &= \pi sin(\pi x)cos(\pi t)+(pi)^2 sin(x\pi)sin(t\pi)+\pi\lambda cos(x\pi)sin(t\pi)(sin(x\pi)sin(t\pi))^{\delta} \\ &- \beta(sin(\pi x)sin(\pi t)(1-(sin(x\pi)sin(t\pi))^{\delta}). \end{aligned}$$

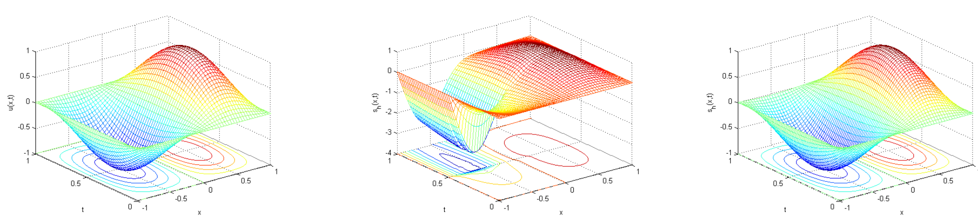
The exact solution is

$$u(x,t)=sin(x\pi)sin(t\pi) \qquad (x,t)\in[-1,1]\times[-1,1]. \tag{4.51}$$



(A) BDFS and exact solutions (B) SSPRK54S and exact solutions

FIGURE 3. Computed solutions of problem (4.49) for $\lambda=0.01, \beta=\gamma=0.001$, $\delta=4$ and $\Delta t=1e-02$



(A) Exact solution (B) SSPRK54S solution (C) BDFS solution

FIGURE 4. Computed solutions of problem (4.49) for $\lambda=0.001$, $\gamma=1$, $\beta=0.001$, $\delta=8$ and $\Delta t=1e-03$

Various relative errors for problem (4.49) by using the BDFS and SSPRK54S methods have been presented in Table 5. Thus, the results obtained by the BDFS method are accurate while the SSPRK54S did not work for large values of δ. For various time values, comparison between the BDFS, SSPRK54S solutions and exact solution is carried out as seen in Figures 3(B) and 3(A). In the figures, we observe that the BDFS and exact solutions are in good agreement. The physical behaviour of the problem (4.49) has been presented in Figure 4. It can be seen that the proposed scheme is in very good agreement with the exact one and exhibits physical characteristics of the problem correctly.

TABLE 5. The relative errors of the GBFEF for the proposed methods

(A)

$\lambda = 0.01,\ \delta = 1,\ \beta = \gamma = 1,\ \Delta t = 1e-3$		
$e_{1,2,\infty}$	SSPRK54S	BDFS
e_1	$2.19e-1$	$8.49e-3$
e_2	$1.44e-1$	$8.55e-3$
e_∞	$1.86e-1$	$7.88e-3$

(B)

$\lambda = 0.001,\ \delta = 4,\ \beta = -0.01,\ \gamma = 1,\ \Delta t = 1e-4$		
$e_{1,2,\infty}$	SSPRK54S	BDFS
e_1	$1.04e-1$	$3.47e-3$
e_2	$1.34e-1$	$4.95e-4$
e_∞	$1.63e-1$	$5.74e-4$

(C)

$\lambda = 0.0001,\ \delta = 500,\ \beta = \gamma = 0.01,\ \Delta t = 1e-4$		
$e_{1,2,\infty}$	SSPRK54S	BDFS
e_1	N.W.	$4.17e-3$
e_2	N.W.	$4.92e-5$
e_∞	N.W.	$4.33e-5$

(D)

$\lambda = 0.0001,\ \delta = 10000,\ \beta = \gamma = 1,\ \Delta t = 1e-5$		
$e_{1,2,\infty}$	SSPRK54S	BDFS
e_1	N.W.	$2.48e-3$
e_2	N.W.	$3.66e-3$
e_∞	N.W.	$3.28e-3$

4.2. **Numerical Behaviour of the GBHEF.** We consider the GBHEF of the form

$$\frac{\partial u}{\partial t} - \frac{1}{Re}\frac{\partial^2 u}{\partial x^2} + \beta u^\delta \frac{\partial u}{\partial x} - \gamma u(1-u^\delta)(u^\delta - C) - f(x,t) = 0, \quad (x,t) \in \Omega = [a,b] \times [t_0,T], \tag{4.52}$$

with the initial and boundary conditions given by

$$u(x,t_0) = u_0(x), \tag{4.53}$$

$$u(a,t) = g_1(t), \quad u(b,t) = g_2(t). \tag{4.54}$$

Here, λ, β, γ and δ are physical constants. In this example, we use the proposed numerical schemes to find an approximation $s_h(x,t)$ to the exact solution $u(x,t)$ given by (2.5). By rearranging equation (4.52) in the form of (1.1), we can define the linear part and nonlinear part (involving the forcing term), respectively as

$$\mathscr{L}(\frac{\partial^2 u}{\partial x^2}, \frac{\partial u}{\partial x}, u, x, t) = \lambda u_{xx} - \gamma C u,$$

and

$$\mathscr{N}(\frac{\partial^2 u}{\partial x^2}, \frac{\partial u}{\partial x}, u, x, t) = -\delta u^\delta u_x + \gamma u u^\delta - \gamma C u - \gamma u u^{2\delta} + \gamma C u u^\delta + f(x,t).$$

Now, from the above parts, we get $a_0 = -\gamma C$, $a_1 = 0$ and $a_2 = \lambda$. Considering the relations (2.14), we obtain

$$\begin{aligned} y_0'(t) &= -\gamma C g_1(t) + F(y(t), x_0, t), \\ y_m'(t) &= -\gamma C g_2(t) + F(y(t), x_m, t), \end{aligned} \tag{4.55}$$

where

$$F(y(t),x_0,t) = \big(\gamma g_1(t) g_1(t))^\delta\big)\big(1 - (g_1(t))^\delta + C\big) - \frac{\beta}{h}(g_1(t))^\delta\big(g_1(t) - y_1(t)\big) + f(x_0,t),$$

and

$$F(y(t),x_m,t) = \big(\gamma g_2(t) g_2(t))^\delta\big)\big(1 - (g_2(t))^\delta + C\big) - \frac{\beta}{h}(g_2(t))^\delta\big(y_{m-1}(t) - g_2(t)\big) + f(x_m,t).$$

Now, from (2.11) and (4.55), by evaluating these equations at points x_1 and x_{m-1}, we obtain

$$\begin{aligned} \frac{4}{6}y_1'(t)+\frac{1}{6}y_2'(t) &= \Big(\frac{-2\gamma C}{3}-\frac{2\lambda}{h^2}\Big)y_1(t)+\Big(\frac{-\gamma C}{6}+\frac{\lambda}{h^2}\Big)y_2(t) \\ &+ \Big(\frac{\lambda}{h^2}\Big)g_1(t)+F(y(t),x_1,t)-\frac{1}{6}F(y(t),x_0,t), \end{aligned} \tag{4.56}$$

and

$$\begin{aligned} \frac{1}{6}y_{m-2}'(t)+\frac{4}{6}y_{m-1}'(t) &= \Big(\frac{\gamma}{6}+\frac{\lambda}{h^2}\Big)y_{m-2}(t)+\Big(\frac{2\gamma}{3}-\frac{2\lambda}{h^2}\Big)y_{m-1}(t)+\Big(\frac{\lambda}{h^2}\Big)g_2(t) \\ &+ F(y(t),x_{m-1},t)-\frac{1}{6}F(y(t),x_m,t). \end{aligned} \tag{4.57}$$

At points x_i, $i=2,\dots,m-2$, we obtain

$$\begin{aligned} \frac{1}{6}y_{i-1}'+\frac{4}{6}y_i'+\frac{1}{6}y_{i+1}' &= \Big(\frac{-\gamma C}{6}+\frac{\lambda}{h^2}\Big)y_{i-1}(t)+\Big(\frac{-2\gamma C}{3}-\frac{2\lambda}{h^2}\Big)y_i(t) \\ &+ \Big(\frac{-\gamma C}{6}-\frac{\lambda}{h^2}\Big)y_{i+1}(t)+F(y(t),x_i,t) \end{aligned}, \tag{4.58}$$

where

$$\begin{aligned} F(y(t),x_1,t) &= \Big(-\gamma\big(g_1(t)+y_1(t)+y_2(t)\big)\big(\frac{1}{6}g_1(t) \\ &+ \frac{4}{6}y_1(t)+\frac{1}{6}y_2(t)\big)^{\delta}\Big)\Big(\big(\frac{1}{6}g_1(t)+\frac{4}{6}y_1(t)+\frac{1}{6}y_2(t)\big)^{\delta}-C\Big) \\ &- \frac{\beta}{2h}\big(\frac{1}{6}g_1(t)+\frac{4}{6}y_1(t)+\frac{1}{6}y_2(t)\big)^{\delta}\big(g_1(t)-y_2(t)\big), \end{aligned}$$

$$\begin{aligned} F(y(t),x_m,t) &= \Big(-\gamma\big(y_{m-2}(t)+4y_{m-1}(t)+g_2(t)\big)\Big(1-\big(\frac{1}{6}y_{m-2}(t)+\frac{4}{6}y_{m-1}(t)+\frac{1}{6}g_2(t)\big)^{\delta}\Big) \\ &\quad \Big(\big(\frac{1}{6}y_{m-2}(t)+\frac{4}{6}y_{m-1}(t)+\frac{1}{6}g_2(t)\big)^{\delta}-C\Big) \\ &- \frac{\beta}{2h}\big(\frac{1}{6}y_{m-2}(t)+\frac{4}{6}y_{m-1}(t)+\frac{1}{6}g_2(t)\big)^{\delta}\big(y_{m-2}(t)-g_2(t)\big), \end{aligned}$$

$$\begin{aligned} F(y(t),x_i,t) &= \Big(-\gamma\big(y_{i-1}(t)+4y_i(t)+y_{i+1}(t)\big)\Big(1-\big(\frac{1}{6}y_{i-1}(t)+\frac{4}{6}y_i(t)+\frac{1}{6}y_{i+1}(t)\big)^{\delta}\Big) \\ &- \frac{\beta}{2h}\big(\frac{1}{6}y_{i-1}(t)+\frac{4}{6}y_i(t)+\frac{1}{6}y_{i+1}(t)\big)^{\delta}\big(y_{i-1}(t)-y_{i+1}(t)\big). \end{aligned}$$

The approximating cubic spline s_h must also satisfy the initial condition (4.34) at points $x_0,\dots,x_m$ and at initial time t_0 :

$$\begin{cases} s_h(x_0,t_0) = u_0(x_0), & \text{for} \quad i=0, \\ s_h(x_i,t_0) = u_0(x_i), & \text{for} \quad i=1,\dots,m-1, \\ s_h(x_m,t_0) = u_0(x_m), & \text{for} \quad i=m. \end{cases} \tag{4.59}$$

By virtue of (2.8), (2.9) and (2.11), we find

$$A\alpha(t_0)=\alpha_0, \tag{4.60}$$

where $y_0=[6u_0(x_1)-g_1(t_0),6u_0(x_2),\dots,6u_0(x_{m-2}),6u_0(x_{m-1})-g_2(t_0)]^T$ and the matrix A of size $(m-1)\times(m-1)$ is given by (2.21). Now, equations (4.37), (4.38), (4.39) and (4.41) are expressed compactly as the following system of ordinary differential equations

$$\left\{\begin{array}{lcl} A\dfrac{dy(t)}{dt} & = & Dy(t)+\Phi(y(t)), \\ \\ Ay(t_0) & = & y_0, \end{array}\right. \tag{4.61}$$

where the matrix D of size $(m-1)\times(m-1)$ is

$$D=\begin{pmatrix} d_0 & d_1 & 0 & \cdots & 0 \\ d_1' & d_0 & d_1 & & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & d_1' & d_0 & d_1 \\ 0 & \cdots & 0 & d_1' & d_0 \end{pmatrix}, \tag{4.62}$$

where $d_0=\dfrac{-2\gamma C}{3}-\dfrac{2\lambda}{h^2}$, $d_1=\dfrac{-\gamma C}{6}+\dfrac{\lambda}{h^2}$, and $d_1'=\dfrac{-\gamma C}{6}+\dfrac{\lambda}{h^2}$. The vector valued function Φ is given by $\Phi(y(t))=[\Phi_1(y(t)),\Phi_2(y(t)),\ldots,\Phi_{m-2}(y(t)),\Phi_{m-1}(y(t))]^T$ with

$$\Phi_1(y(t))=\big(\frac{\lambda}{h^2}\big)g_1(t)+F(y(t),x_1,t)-\frac{1}{6}F(y(t),x_0,t),$$
$$\Phi_{m-1}(y(t))=\big(\frac{\lambda}{h^2}\big)g_2(t)+F(y(t),x_{m-1},t)-\frac{1}{6}F(y(t),x_m,t),$$

and

$$\Phi_i(y(t))=F(y(t),x_i,t)\ \text{ for }\ i=2,\ldots,m-2.$$

Now, we solve the system (4.61) by using the BDFS method, as given in the previous section.

4.2.1. *Example 3.* We consider the GBHE of the form of equation (4.52) with the exact solution given by

$$u(x,t)=\Big(\frac{C}{2}+\frac{C}{2}tanh\big[A_1\big(x-A_2t\big)\big]\Big)^{1/\delta}, \tag{4.63}$$

where

$$A_1=\frac{-\beta\delta+\delta\sqrt{\beta^2+4\gamma(1+\delta)}}{4(1+\delta)}C,$$

and

$$A_2=\frac{C\beta}{(1+\delta)}-\frac{(1+\delta-C)(-\beta+\sqrt{\beta^2+4\beta(1+\delta)})}{2(1+\delta)}.$$

The initial condition is given by

$$u(x,0)=\Big(\frac{C}{2}+\frac{C}{2}tanh\big(A_1x\big)\Big)^{1/\delta}=u_0(x), \tag{4.64}$$

where β, γ, δ, λ are constant parameters and $C=0.1$. Numerical solutions of this problem are obtained by taking δ as 1, 8, 500, 10000 for various values of β, γ and λ in the domain $(x,t)\in\Omega=[-1,1]\times[0,1]$. In Table 6, the accuracy of the proposed schemes is examined by computing the relative errors for large values of δ and smaller values of λ. Here, it can be noted that the BDFS results are in good agreement with the exact solution for large values of δ while the SSPRK54S did not worked (N.W.). The relative and absolute errors are documented in Table 7 and are compared with some previous works. We have seen from the corresponding table that the errors obtained by the BDFS and SSPRK54S schemes are quite small and furthermore, better than most of the methods available in the literature. The BDFS and SSPRK54S solutions with exact solution of this problem are plotted in Figure 5. It can be deduced that the BDFS scheme solutions are very compatible with the exact solutions and furthermore, better than the SSPRK54S scheme. The physical behavior of the solutions is presented in Figure 6. Note that behaviour of the BDFS is in good agreement with exact solution at free of choice of the physical parameters.

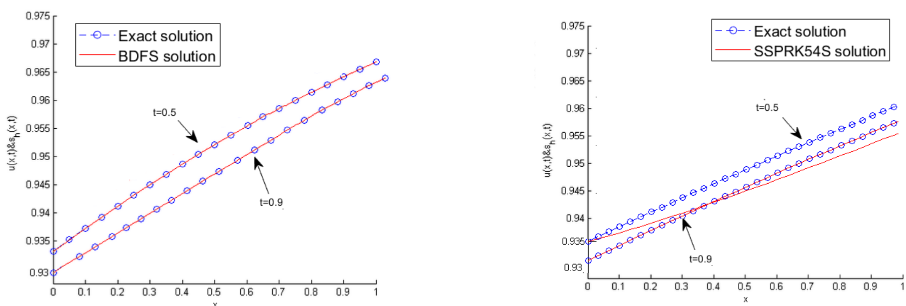
TABLE 6. The relative errors of the GBHE for the proposed methods

(A)

$\lambda = 0.001,\ \beta = \gamma = 1000, \delta = 500$ $\Delta t = 1e-4\ ,\ C = 0.0001$		
$e_{1,2,\infty}$	SSPRK54S	BDFS
e_1	N.W.	$7.94e-3$
e_2	N.W.	$8.07e-3$
e_∞	N.W.	$1.08e-2$

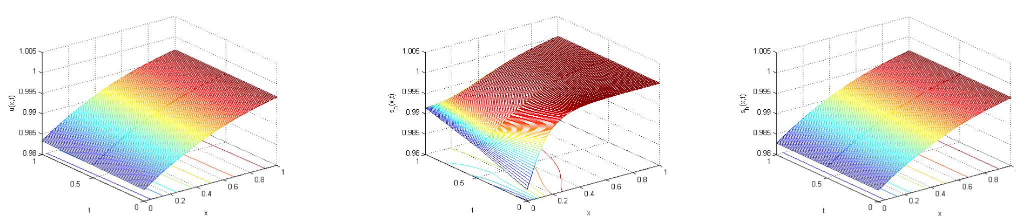
(B)

$\lambda = 0.0001\ \beta = 1,\ \gamma = 5, \delta = 10000$ $\Delta t = 1e-4\ ,\ C = 1$		
$e_{1,2,\infty}$	SSPRK54S	BDFS
e_1	N.W.	$4.00e-4$
e_2	N.W.	$2.33e-4$
e_∞	N.W.	$3.15e-5$



(A) BDFS and exact solutions (B) SSPRK54S and exact solutions

FIGURE 5. Computed solutions of problem (4.63) for $\lambda = 1$, $\beta = 0.01$, $\gamma = 0.5$, $\delta = 80$ and $\Delta t = 1e-04$ with $h = 0.02$



(A) Exact solution (B) SSPRK54S solution (C) BDFS solution

FIGURE 6. Computed solutions of problem (4.63) for $\lambda = 0.01$, $\beta = \gamma = 0.1, \delta = 40$, $\Delta t = 1e-3$ and $C = 0.1$ with $h = 0.002$

TABLE 7. Comparison of the errors for Example 3

(A)

$\lambda = \delta = 1, \beta = \gamma = 0.001, C = 0.001$					
	SSPRK54S	BDFS	[20]	[24]	[25]
Errors	$\Delta t = 1e-3$		$\Delta t = 1e-4$		
e_1	$5.10e-5$	$6.67e-7$	- - -	- - -	- - -
e_2	$5.22e-5$	$6.90e-7$	- - -	- - -	- - -
e_∞	$7.63e-6$	$8.91e-8$	- - -	- - -	- - -
L_∞	$6.73e-11$	$1.02e-17$	$1.93e-7$	$3.74e-8$	$4.26e-17$

(B)

$\delta = 8, \lambda = \gamma = 1, \Delta t = 1e-4$					
	SSPRK54S	BDFS	[20]	[24]	[25]
Errors	$C = 0.01, \beta = 80$		$C = 0.0001, \beta = 100$		
e_1	$1.07e-2$	$1.55e-4$	- - -	- - -	- - -
e_2	$1.16e-2$	$5.85e-5$	- - -	- - -	- - -
e_∞	$1.43e-2$	$5.64e-5$	- - -	- - -	- - -
L_∞	$2.03e-5$	$4.82e-8$	$4.58e-8$	$1.27e-8$	$5.55e-17$

5. CONCULUSIONS AND RECOMMENDATIONS

Since the linearization of nonlinear systems loses their real features, with keeping real features of the nature, this article concentrates primarily on numerically capturing its physical response governed by the advection-diffusion-reaction equation with forcing mechanism. In the investigation of the real-world processes without losing their properties, this article has presented two newly combined methods; the backward differentiation formula-spline (BDFS) and the optimal five stage and fourth-order strong stability preserving Runge-Kutta-spline (SSPRK54S) methods. In the current methods, it has been importantly concluded that neither linearization effort to deal with nonlinear terms nor transforming the process is required. Notice that the current methods have been figured out to be more effective than the literature for the problem of interest. The computed results have revealed that the BDFS method is more accurate and computationally more economical in comparison with the SSPRK54S method. The first method has been realized to be more reliable than the latter, even a very important alternative for the research society, in analysing the problem by conserving the physical properties of the nature. The results showed that the BDFS scheme is relatively free of choice of the physical parameters. In the forthcoming studies, with the use of same mechanisms, behaviour of the ADR processes with forcing terms can be numerically taken into consideration in the higher dimensional domains.

REFERENCES

[1] Behzadi, S. S., *Numerical solution for solving Burgers-Fisher equation by iterative methods*, Math. Comput. Appl., **6** (2011), 443–455

[2] Biazar, J. and Muhammadi, F., *Application of diferential transform method to the generalized Burgers-Huxley equation*, Appl. Appl. Math., **5** (2010), 1726–1740

[3] Brajesh, K. S., Geeta, A. and Manoj, K. S., *A numerical scheme for the generalized Burgers-Huxley equation*, J. Egyptian Math. Soc., **24** (2016), 629–637

[4] Bukhari, M., Arshad, S. B. and Saqlain, S. M., *Numerical solution of generalized Burgers-Huxley equation using local radial basis functions*, International Journal of Advanced and Applied Sciences **4** (2017), 1–11

[5] Carl, D. B., *Partial differential equations of parabolic type*, Prentice-Hall INC NEW Jersey, 1964

[6] Carl, D. B., *A Practical Guide to Spline*, Mathematics of Computation, **27** (1978)

[7] Coroian, I., *Some Runge-Kutta type formulas with large region of absolute stability*, Carpathian J. Math., **5** (1980), 10–15

[8] Dag, I., Irk, I. and Sari, M., *The extended cubic B-spline algorithm for a modified regularized long wave equation*, Chinese Physics B **22** (2013), 1674–1056

[9] Dag I., Irk, D., Sahin, A., *B-spline collocation methods for numerical solutions of the Burgers equation*, Mathematical Problems in Engineering, **5** (2005), 521–538

[10] Dag I., Ersoy O., *Numerical solution of generalized Burgers-Fisher equation by exponential cubic B-Spline collocation method*, In AIP Conference Proceedings: Proceedings of the International Conference on Numerical Analysis and Applied Mathematics, **1648** (2015), 370-008

[11] Darvishi, M. T., Kheybari, S. and Khan, F., *Spectral collocation method and Darvishis preconditionings to solve the generalized Burgers-Huxley equation*, Communication in Nonlinear Scince and Numerical Simulation, **13** (2008), 2091–21013

[12] Erdogan, U., Sari, M. and Kocak, H., *Efficient numerical treatment of nonlinearities in the advection-diffusion-reaction equations*, International Journal of Numerical Methods for Heat & Fluid Flow, **29** (2019), 1–20

[13] Fisher, R. A., *The wave of advance of advantageous genes*, Annals of Eugenics **1** (1937), 353–369

[14] Golbabai, A. and Javidi, M., *A spectral domain decomposition approach for the generalized Burgers-Fisher equation*, Chaos Solutions Fractals, **39** (2009), 385–392

[15] Goh, J., Majid, A. A. and Ismail, A. I. Md., *Cubic B-spline collocation method for one-dimensional heat and advection-diffusion equations*, J. Appl. Math., 2012, Art. ID 458701, 8 pp.

[16] Hammad, D. A., El-Azab M. S., *2N order compact fnite diference scheme with collocation method for solving the generalized Burgers-Huxley and Burgers-Fisher equations*, Appl. Math. Comput., **258** (2015), 296–311

[17] Hashim, I., Noorani, S. M. and Said Al-Hadidi, M. R., *Solving the generalized Burgers-Hulxery equation using the a domain decomposition method*, Math. Comput. Modelling, **43** (2006), 1404–1411

[18] Imtiaz, W., Muhammad A. and Muhammad, A., *Hybrid B-Spline Collocation Method for Solving the Generalized Burgers-Fisher and Burgers-Huxley Equations*, Mathematical Problems in Engineering, **18** (2018), 1–18

[19] Inan, B. and Bahadir, A. R., *Numerical solutions of the generalized Burgers-Huxley equation by implicit exponential fnite diference method*, J. Appl. Math. Stat. Inform., **11** (2015), 1–17

[20] Ismail, H. N. A. and Raslan, K., *Raslan K. A domain decomposition method for Burgers-Huxley and Burgers-Fisher equations*, Appl. Math. Comput., **159** (2004), 291–301

[21] Kolmogrov, A., Petrovski, I. and Piskunov, N. A, *A study of the equation of diffusion with increase in the equation of matter and its application to a biological problem*, Bjul, Moskov Skogogos, 1937

[22] Lyche, T. and Schumaker, L., *A Multi resolution Tensor Spline Method for Fitting Functions on the Sphere*, SIAM J. Sci. Comput., **22** (2009), 724–746

[23] Lyche, T., *Knot Insertion and Deletion Algorithms for B-Spline Curves and Surfaces*, Society for Industrial and Applied Mathematics, 1987

[24] Mittal, R. C., *Tripartite A. Numerical solutions of generalized Burgers-Fisher and generalized Burgers-Huxley A compact finite difference method for the generalized Burgers-Fisher equation*, Int. J. Comput. Math., **92** (2015), 1053–1077

[25] Mohanty, R. and Sharma, S., *High-accuracy quasi-variable mesh method for the system of 1D quasi-linear parabolic partial differential equations based on off-step spline in compression approximations*, Advances in Difference Equations 2017, 2017:212, https://doi.org/10.1186/s13662-017-1274-3

[26] Reza, M., *B-spline collocation algorithm for numerical solution of the generalized Buegers-Huxley equation, Numerical Methods for Partial Diferential Equations*, Numerical Methods for Partial Diferential Equations **29** (2013), 1173–1191

[27] Sanda, M., *On spline collocation and the Hilbert transform*, Carpathian J. Math., **31** (2015), 89–95

[28] Sari, M. and Gurarslan, G., *Numerical solutions of the generalized Burgers-Huxley equation by a differential quadrature method*, Mathematical Problems in Engineering, **11** (2009), 370–765

[29] Sari, M., Gurarslan, G. and Dag, I., *A compact finite difference method for the generalized Burgers-Fisher equation*, Numerical Methods Partial Differential Equations, **26** (2010), 438–445

[30] Schultz, M. H., *Spline analysis*, Prentice-Hall. ING, 1973

[31] Schumaker L., *Spline Functions, Basic Theory*, Cambridge Mathematical Library, 2007

[32] Singh, B. K. and Arora, G., *A numerical scheme to solve Fisher-type reaction-diffusion equations*, Mathematics in Engineering, Science and Aerospace (MESA), **2** (2014), 153–164

[33] Singh, B. K., Arora, G. and Singh, M. K., *A numerical scheme for the generalized Burgers-Huxley equation*, J. Egyptian Math. Soc., **24** (2016), 629–637

[34] Shiralashetti, S. C. and Kumbinarasaiah, S., *Cardinal B-Spline Wavelet Based Numerical Method for the Solution of Generalized Burgers-Huxley Equation*, International Journal of Applied and Computational Mathematics **4** (2018), 4–73

[35] Spiteri, R. J. and Ruuth, S. J., *A new class of optimal high-order strong-stability-preserving time discretization methods*, SIAM J. Numer. Anal., **40** (2002), 469–491

[36] Suheel, A. M., Ijaz, M. Q., Muhammad, A. and Aqdas, N. M., *Numerical solution to generalized Burgers-Fisher equation using exp-function method hybridized with heuristic computation*, Plod One, **10** (2015), 1–15

[37] Vladimir, P., *Runge-Kutta method for second order ordinary differential equation*, Carpathian J. Math., **12** (1996), 235–241

[38] Yanagida, E., *Standing Pulse Solutions in Reaction-Diffusion Systems with Skew Gradient Structure*, J. Dynam. Differential Equations, **10** (2002), 189–205

[39] Zhang, L., Wang, L. and Ding, X., *Exact finite difference scheme and nonstandard finite difference scheme for Burgers and Burgers-Fisher equations*, J. Appl. Math., **2** (2014), 1–12.

[40] Zhu, C. and Kang, W. S., *Numerical solution of Burgers-Fisher equation by cubic B-spline quasi-interpolation*, Appl. Math. Comput., **216** (2010), 2679–2686

[1]DEPARTMENT OF MATHEMATICS
FACULTY OF ARTS AND SCIENCE
YILDIZ TECHNICAL UNIVERSITY
34220, ISTANBUL, TURKEY
E-mail address: sarim@yildiz.edu.tr
E-mail address: shko.ali.tahir@std.yildiz.edu.tr

[2]DEPARTMENT OF MATHEMATICS
UNIVERSITÉ DU LITTORAL CÔTE D'OPALE
50 RUE F. BUISSON
BP 699, 62228 CALAIS-CEDEX, FRANCE
E-mail address: bouhamidi@lmpa.univ-littoral.fr